

CLAIM AMENDMENTS

1. (Currently Amended) A method for calculating the jitter of a packet flow sent by a program on a sending computer over a network to a receiving computer, comprising:
 - accessing data collected on the sending computer, said data comprising identifiers of a plurality of packets sent by the program along with timestamps representing the times of transmission of the sent packets;
 - accessing data collected on the receiving computer, said data comprising identifiers of a plurality of packets received from the network along with timestamps representing the times of reception of the received packets;
 - associating, through the use of the sent and received packet identifiers, at least some of the sent packets with received packets by comparing a field uniquely identifying the packet flow in the sent and received packet identifiers and comparing an IP ID assigned to the packet in the sent and received packet identifiers; and
 - calculating jitter as the variation in the differences between the reception and transmission timestamps of associated packets.
2. (Original) The method of claim 1 wherein associating and calculating overlap in time with accessing data collected on the sending and receiving computers.
3. (Original) The method of claim 1 wherein accessing data collected on the sending computer comprises sending said data to the receiving computer.
4. (Original) The method of claim 1 wherein accessing data collected on the receiving computer comprises sending said data to the sending computer.

5. (Original) The method of claim 1 wherein accessing data collected on the sending and receiving computers comprises sending said data to a computer other than the sending and receiving computers.
6. (Canceled)
7. (Currently Amended) The method of claim 6 wherein comparing ~~of~~ a field identifying the packet flow comprises comparing a sender IP address, a sender port, a receiver IP address, a receiver port, and a protocol identifier.
8. (Canceled)
9. (Original) The method of claim 1 wherein associating at least some of the sent packets with received packets comprises noting as lost in transmission sent packets which are not associated with received packets.
10. (Original) The method of claim 1 wherein associating at least some of the sent packets with received packets comprises using the received packet identifiers to reorder the data collected on the receiving computer into the order in which the received packets were sent.
11. (Original) The method of claim 10 wherein reordering comprises comparing a rollover component of the received packet identifiers.
12. (Original) The method of claim 11 wherein reordering comprises imposing a window on the range of possible values of the rollover component of the received packet identifiers and reordering only the data the values of whose rollover component are within the window.

13. (Original) The method of claim 12 wherein reordering comprises imposing a window that is smaller than the range of possible values of the rollover component of the received packet identifiers, reordering only the data the values of whose rollover component are within the window, and moving the window throughout the range until all the data are reordered.
14. (Original) The method of claim 11 wherein associating comprises imposing a window on the range of possible values of the rollover component of the received packet identifiers and searching for a received packet identifier to match a sent packet identifier only among those data the values of whose rollover component are within the window.
15. (Original) The method of claim 14 wherein associating comprises imposing a window that is smaller than the range of possible values of the rollover component of the received packet identifiers, searching for a received packet identifier to match a sent packet identifier only among those data the values of whose rollover component are within the window, and moving the window throughout the range until all sent packets are either associated with received packets or are noted as lost in transmission.
16. (Original) The method of claim 1 wherein calculating the jitter comprises correcting for jumps in the clocks on the sending and receiving computers.
17. (Original) The method of claim 1 wherein calculating the jitter comprises correcting for skew between the clocks on the sending and receiving computers.

18. (Original) A computer-readable medium containing instructions for performing the method of claim 1.
19. (Currently Amended) A computer-readable medium having stored thereon a data structure, comprising:
 - a first data field containing data representing an ~~identify~~ identity of a packet flow;
 - a second data field containing data representing an ~~identify~~ identity of a packet transmitted in the packet flow;
 - a third data field containing data representing a time of transmission of the packet;
 - and
 - a fourth data field containing data representing a time of reception of the packet.
20. (Original) The computer-readable medium of claim 19 wherein the second field comprises a fifth field containing data representing an IP ID assigned to the packet.
21. (Original) The computer-readable medium of claim 19 wherein the first field comprises:
 - a fifth field containing data representing a sender IP address of the packet flow;
 - a sixth field containing data representing a sender port of the packet flow;
 - a seventh field containing data representing a receiver IP address of the packet flow;
 - an eighth field containing data representing a receiver port of the packet flow;
 - and
 - a ninth field containing data representing a protocol identifier of the packet flow.

22. (Original) A computer-readable medium having stored thereon data structures according to claim 19, the data structures sorted into the order of the times of transmission of the packets.
23. (New) The method of claim 1 further comprising normalizing send timestamps and receive timestamps to account for a clock skew effect.
24. (New) The method of claim 1 further comprising analyzing the timestamps to determine whether a timer jump has occurred.